LiDDM: A Data Mining System for Linked Data

Venkata Narasimha Pavan Kappara
Indian Institute of Information Technology Allahabad
Allahabad, India
kvnpavan@gmail.com

Ryutaro Ichise
National Institute of Informatics
Tokyo, Japan
ichise@nii.ac.jp

O.P. Vyas
Indian Institutes of Information Technology Allahabad
Allahabad, India
opvyas@iiita.ac.in

ABSTRACT

In today's scenario, the quantity of linked data is growing rapidly. The data includes ontologies, governmental data, statistics and so on. With more and more sources publishing the data, the amount of linked data is becoming enormous. The task of obtaining the data from various sources, integrating and fine-tuning the data for desired statistical analysis assumes prominence. So there is need of a good model with efficient UI design to perform the Linked Data Mining. We proposed a model that helps to effectively interact with linked data present in the web in structured format, retrieve and integrate data from different sources, shape and fine-tune the so formed data for statistical analysis, perform data mining and also visualize the results at the end.

1. INTRODUCTION

Since the revolution of linked data, the amount of data that is being available in the web in structured format in the cloud of linked data is growing at a very fast pace. LOD (Linking Open Data) forms the foundation for linking the data available on the web in structured format. This community helps to link the data published by various domains as companies, books, scientific publication, films, music, radio program, genes, clinical trial, online communities, statistical and scientific data [3]. This community provides different datasets in RDF(Resource Description Framework) format and also provides RDF links between these datasets that enables us to move from one data item in one dataset to other data item in other data set. There are number of organizations that are publishing their data in the linked data cloud in different domains. Linked data, as we look at it today, is very complex and dynamic pertaining to its heterogeneity and diversity.

Various datasets available in the Linked Data Cloud has their own significance in terms of their usability. In today's scenario the result related to user query for extracting a useful hidden pattern may not always be completely answered by using only one (or many) of the dataset in isolation. Here linked data comes into picture as there is a need to integrate different data sources available in different structured format to answer such type of complex queries. If you look at data sources like World FactBook [5], Data.gov [16], DBpedia [2], the data that they provide is real world data. The information that these kinds of data provide can be helpful in many ways such as predicting the future outcome given the past statistics, the dependency of one attribute over another attribute and so on. In this context, it is necessary to extract hidden information from the linked data considering its richness of information.

Our proposed model suggest a Framework tool for Linked Data Mining that capture data from linked data cloud and extract various interesting hidden information. This model is targeted to deal with the complexities associated with mining the linked data efficiently. Our hypothesis is implemented in form of a tool that takes the data from linked data cloud, performs various KDD(Knowledge Discovery in Databases) operations on linked data and applies data mining technique such as association, clustering etc. and also visualizes the result at the end.

The remaining sections are organized as follows. The second section deals with back ground and related work. The third section describes the architecture of LiDDM(Linked Data Data Miner). The fourth section discusses the tool that we made to implement the model. The fifth section deals with the case study. The sixth section comes up with discussions and future work. Finally the seventh section is the conclusion.

2. RELATED WORK

Linked data refers to a set of best practices for publishing and connecting structured data on the web [3]. With the expansion of Linking Open Data Project, more and more data available on the web are getting converted into RDF and getting published as linked data. The difference between interacting with a web of data and a web of documents has been discussed in [11]. This web of data is richer in information and is also available in standard format. Therefore, to exploit the hidden information in this kind of data, we have to first understand the related work done previously. Looking at the general process of KDD, the steps in the process of knowledge discovery in databases have been explained [8]. The data has to be selected, preprocessed, transformed, mined, evaluated and interpreted for the process of Knowledge Data Discovery [8]. For the process of knowledge

Copyright is held by the author/owner(s).
LDOW2011, March 29, 2011, Hyderabad, India.

discovery in the semantic web, SPARQL-ML was introduced by extending the given SPARQL language to work with statistical learning methods [12]. This imposes the burden of having the knowledge of extended SPARQL and its ontology on the users. Some researches [14] have extended the model for adding data mining method to SPARQL [18] by relieving the burden on users to have the exact knowledge of ontological structures by asking them to specify the context to automatically retrieve the items that form the transaction. However, ontology axioms and semantic annotations for the process of association rule mining have been used earlier [14].

In our approach, we modified the model used by U. Fayyad et al [8], which is general process of KDD, to suit the needs of linked data. Instead of extending SPARQL [18], we retrieved the linked data using normal SPARQL queries and instead focused on the process of refining and weaving the retrieved data to finally transform it to be fed into the data mining module. This approach separated the work of retrieving data from the process of data mining and relieved the users from the burden of learning extended SPARQL and its ontology. Also this separation allowed more flexibility in choosing whatever data we needed from various data sources first and then concentrating on mining the data once all the data needed had been retrieved, integrated and transformed. Also LiDDM works by finding classifications and clustering in addition to finding associations.

3. LIDDM: A MODEL

To start with, our model modified the process of KDD, as we discussed in the previous section to conform to the needs of linked data and proceeded in a hierarchical manner. A data mining system was used for statistical analysis and linked data from the linked data cloud was retrieved, processed and fed onto it. Figure 1 provides the overview of our model.

3.1 Data Retrieval through Querying

In this initial step of LiDDM, the data from linked data cloud is queried and retrieved. This step can be compared to the data selection step in the KDD Process. The data retrieved will be in the form of a table with some rows and columns. The rows denote instances of data retrieved and the columns denote the value of each attribute for each instance.

3.2 Data Preprocessing

Once the data retrieval is done, data preprocessing comes into picture which plays a significant role in data mining process. Most of the time data is not in a format suitable for immediate application of data mining techniques. This step highlights that data must be appropriately preprocessed before going for further stages of knowledge discovery.

3.2.1 Data Integration

In the previous step of Linked Data Mining, data is retrieved from multiple data sources existing in Linked data cloud. This allows the feasibility of having distributed data. This data must be integrated in order to provide answer to user's query. Data is integrated based on some common relation presented in respected data sources. Data sources are selected depending on different factors a user wants to study in different sources. For example, if we want to study the effect of growth rate of each country on its film production, data sources selected can be the World FactBook and the Linked Movie Data Base [10]. We can first query the World FactBook for the growth rate of each country. Then we can query the Linked Movie Data Base for information regarding film production of each country and now we have to integrate both the results in order to find the answer of respected query.

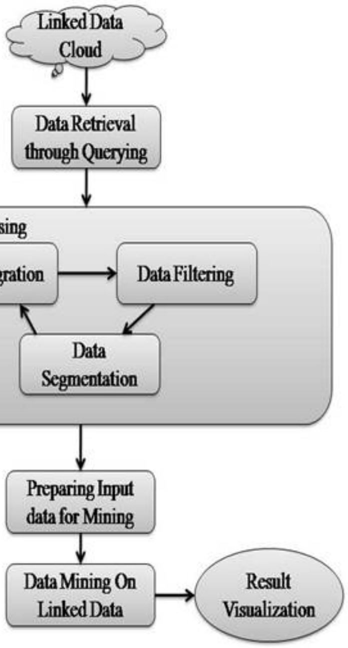


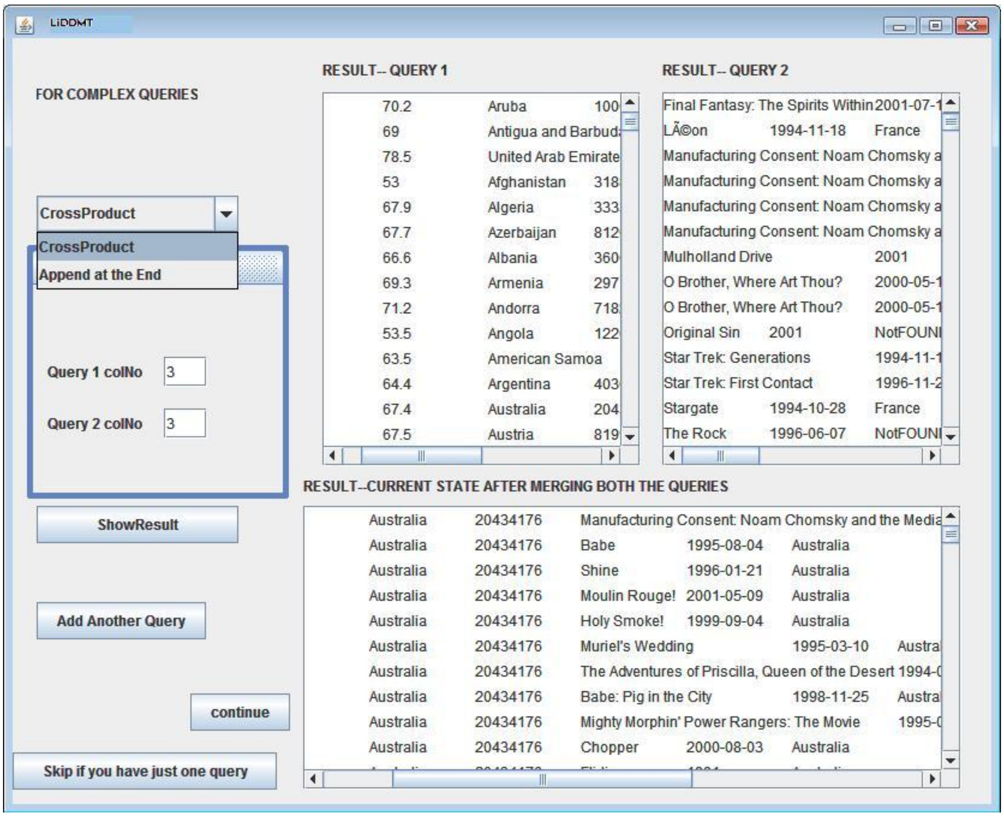
Figure 1: Architecture of LiDDM

3.2.2 Data Filtering

In this step, data that is retrieved and integrated is filtered. Some rows or columns or both are deleted if necessary. Filtering eliminates the unwanted and unnecessary data. For example, let's consider the previous case of the World FactBook and Linked Movie Data Base. If we want the growth rate of a country to be not less than a certain minimum value for research purposes, we can eliminate instances with growth rates less than a certain minimum value at this step.

3.2.3 Data Segmentation

The main purpose of segmenting the data is to divide the data in each column into some classes if necessary for statistical analysis. For example, the data in a certain range can be placed into some class if necessary. Consider the attribute 'population of a country'. In this case, populations less than 10,000,000 can be placed under the segment named 'Low population'. Populations from 10,000,000 to 99,999,999 can be placed under the segment named 'Average Population' and populations from 100,000,000 to 999,999,999 can be placed under the segment named 'High Population'. The step of segmentation step divides the data into different classes and segments, for a class based statistical analysis at the end.



Query 1 colNo 3

Query 2 colNo 3

RESULT-- QUERY 1

```
      70.2          Aruba            100
      69            Antigua and Barbud
      78.5          United Arab Emirate
      53            Afghanistan      318
      67.9          Algeria          333
      67.7          Azerbaijan       812
      66.6          Albania          360
      69.3          Armenia          297
      71.2          Andorra          718
      53.5          Angola           122
      63.5          American Samoa
      64.4          Argentina        403
      67.4          Australia        204
      67.5          Austria          819
```

RESULT-- QUERY 2

```
Final Fantasy: The Spirits Within2001-07-1
LÃ©on          1994-11-18      France
Manufacturing Consent: Noam Chomsky a
Manufacturing Consent: Noam Chomsky a
Manufacturing Consent: Noam Chomsky a
Manufacturing Consent: Noam Chomsky a
Mulholland Drive                2001
O Brother, Where Art Thou?      2000-05-1
O Brother, Where Art Thou?      2000-05-1
Original Sin    2001            NotFOUNI
Star Trek: Generations          1994-11-1
Star Trek: First Contact        1996-11-2
Stargate        1994-10-28      France
The Rock        1996-06-07      NotFOUNI
```

RESULT--CURRENT STATE AFTER MERGING BOTH THE QUERIES

```
      Australia     20434176     Manufacturing Consent: Noam Chomsky and the Media
      Australia     20434176     Babe            1995-08-04     Australia
      Australia     20434176     Shine           1996-01-21     Australia
      Australia     20434176     Moulin Rouge!   2001-05-09     Australia
      Australia     20434176     Holy Smoke!     1999-09-04     Australia
      Australia     20434176     Muriel's Wedding               1995-03-10     Austra
      Australia     20434176     The Adventures of Priscilla, Queen of the Desert 1994-(
      Australia     20434176     Babe: Pig in the City          1998-11-25     Austra
      Australia     20434176     Mighty Morphin' Power Rangers: The Movie      1995-(
      Australia     20434176     Chopper         2000-08-03     Australia
```

Figure 2: This UI shows the data retrieved from World FactBook and Linked Movie Data Base Integration

3.3 Preparing Input Data for Mining

More often than not, the format in which we retrieve the linked data is not the correct format that is required for feeding into the data mining system. Therefore, it is necessary to change the format to the one that is required by the data mining system. The step does exactly this work of format conversion. Thus, this step basically does the same as the transformation of data part in the KDD process.

3.4 Data Mining on Linked Data

In this step, the data mining of the already filtered and transformed data is performed. In this step, you can input the data that is in the format accepted by the data mining system from previous step into the data mining system for analysis. Here the data may be classified or clustered or set for finding association rules. After applying these methods, the results are obtained and visualized for interpretation. Thus LiDDM with all the above features, we believe, will ensure a very good and easy to use framework tool not only for interacting with linked data and visualizing the results but also for re-shaping the data retrieved. The next section deals with the implementation of our model in an application.

4. IMPLEMENTATION WORK

4.1 Tool Environment

To test our model LiDDM, we made an application that implements it. This application was called 'LiDDMT: Linked Data Data Mining Tool'. With this tool, we used Jena API [4] for querying remote data sets in the linked data cloud. Weka API [9] was used for the process of data mining. Weka is widely recognized as the unified platform for performing most of the machine learning algorithms in a single place. Jena is a java framework for building semantic web applications. The tool was made using Java in a Net Beans environment.

4.2 Working of the Tool

Step 1. This tool emulates our model in the following ways. It has a UI for querying the remote data sets. There are two types of querying that are allowed in this model. One is that the user can specify the SPARQL endpoint and SPARQL query for the data to be retrieved. The second type of querying is an automatic query builder that reduces the burden on the user. The possibility of using sub graph patterns for generating automatic RDF queries has been discussed [7]. Our query builder gives the user all the possible predicates he can use given the SPARQL endpoint and asks him to specify only the triples and returns the constructed query. The Drupal Sparql Query Builder [17] also asks the user to specify triples.

Step 2. Regarding Step 2 of our model, which is integration of data retrieved, our tool implements a UI that uses a JOIN operation to perform the JOIN of the retrieved results from two or more queries. It also uses an 'append at the end' operation, which adds the results of two or more queries. Figure 2 shows this functionality. In this figure the text area under 'Result-Query1' gives the results of Query 1, which is a query from the World FactBook and the text area under 'Result-Query2' gives the results of Query 2, which is a query from the Linked Movie Data Base. The text area under 'RESULT-CURRENT STATE AFTER MERGING BOTH THE QUERIES' gives the result of the JOIN operation performed between the 3rd column of query 1 and the 3rd column of query 2 as shown in the figure. Once merging is done, clicking the 'Add another Query' button gives you the option to add a third query. Clicking 'Continue' takes you to Step 3.

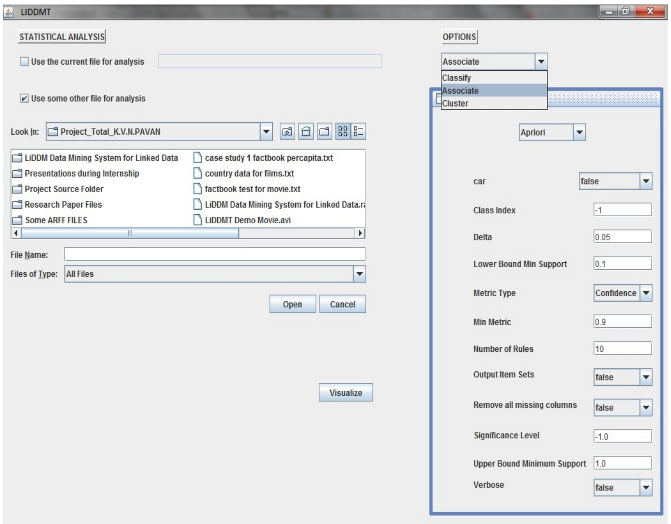
Step 3. Now moving to Step 3 of our model, our tool implements a UI, which is named 'Filter' that filters and cleans the data thus retrieved and integrated. This UI has features of removing unwanted columns, deleting the rows that have values out of a certain range in a numerical column, deleting the rows that have certain strings in certain columns, etc.

Step 4. Now after filtering the data, we move onto UI for Step 4 of our model, which is the segmentation of data. It asks for the name of the segment, and if the values in the column are numeric, we can specify the interval of values that comes in that segment. If the values in the column are string based, then we can specify the set of strings that comes in that segment. Thus our UI converts the data into segments or classes as desired by us for the data mining algorithms to work on it.

Step 5. The UI for Step 5 of our model performs the task of writing the data into the format as required for mining. We used Weka in our tool, and Weka accepts input data in the ARFF(Artribute-Relation File Format) format [13]. Thus this UI asks for the relation name and also the values of attributes for conversion to ARFF format. Once you have finished this conversion, the linked data retrieved becomes acceptable to use for data mining applications using Weka.

Step 6. Our tool has a very flexible UI for data mining (Step 6) in that it has a separate UI for using the original Weka with its full functionality. It also has a simplified version of the UI that is for quick mining where we have implemented the J48 decision tree classification [15], Apriori association [1], and EM (estimation maximization) clustering [6]. Figure 3 shows the simplified version of the data mining tool. Using this UI, you can perform data mining for the ARFF file that was made in Step 5; also you have a file chooser that accepts any other already formed ARFF files, which can also be input for mining and results can be compared and visualized at the same time. In our simplified version of the UI for mining, you can specify the most common options for each of the methods (J48, Apriori, and EM) and can cross check the results by varying different parameters.

Views of Results. The results that are output from this step are visualized at the end. The results from the J48 decision tree classifier are visualized in the form of a decision tree along with classifier output like precision recall, F-Measure etc. Similarly, the results from EM clustering are visualized in the form of an X-Y plot with clusters shown. The results from Apriori association if any, can be visualized in the form of printing the best associations found.



□ Use the current file for analysis

☑ Use some other file for analysis

Look In: Project_Total_K.V.N.PAVAN

LiDDM Data Mining System for Linked Data
Presentations during Internship
Project Source Folder
Research Paper Files
Some ARFF FILES
case study 1 factbook percapita.txt
country data for films.txt
factbook test for movie.txt
LiDDM Data Mining System for Linked Data.ra
LiDDMT Demo Movie.avi

File Name:
Files of Type: All Files

Associate
Apriori

car: false
Class Index: -1
Delta: 0.05
Lower Bound Min Support: 0.1
Metric Type: Confidence
Min Metric: 0.9
Number of Rules: 10
Output Item Sets: false
Remove all missing columns: false
Significance Level: -1.0
Upper Bound Minimum Support: 1.0
Verbose: false

Figure 3: This UI shows the simplified version of data mining tool.

Also as described in our model, our tool LiDDMT has forward and backward moment flexibility in Step 3, Step 4 and Step 5 i.e.; in filter, segmentation and writer, where you can get the results at any step and can go back and forth to any other step. The same is the case with Step 2(in the model) where even with our tool, the UI allows integration of any number of queries as long as they can be merged using either the 'JOIN' operation or 'append at the end' operation.

5. CASE STUDY

Our tool LiDDMT has been tested with many datasets like DBpedia, Linked Movie Data Base, World FactBook, Data.gov etc. However, here for the process of explanation, we choose to demonstrate the effectiveness of our tool from the experiments with the World FactBook dataset.

World FactBook dataset provides information on the history, people, government, economy, geography, communications and other transnational issues of every country for about 266 world entities. We explored this dataset and found out some interesting patterns using our tool.

First, we queried the World FactBook Database for GDP per capita, GDP composition by agriculture, GDP composition by industry, and GDP composition by services of every country. Then in step 4, which is segmentation, we divided each of the attributes, i.e. GDP by agriculture, industry, and services, into 10 classes each at equal intervals of 10 percent. Then GDP per capita is divided into three classes called low, average, and high depending on whether the value is less than 10,000, between 10,000 and 25,000, or more than 25,000 respectively. This segmented data is sent as input to the Apriori algorithm, and we found two association rules that have proved to be very accurate. The rules are as follows:

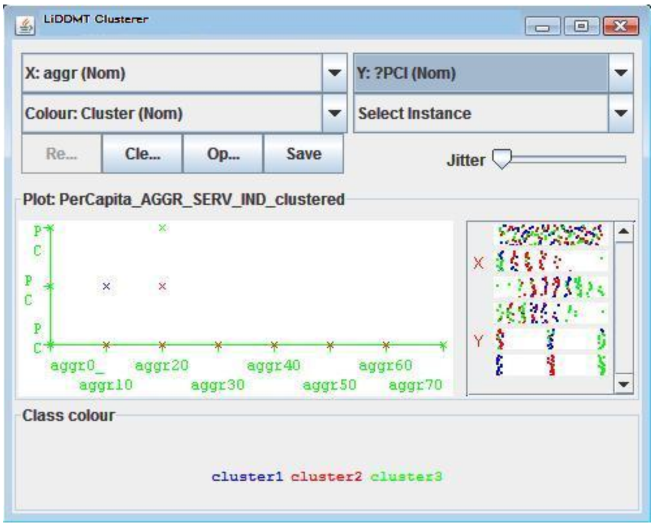


Figure 4: Here PC denotes GDP per capita and aggr-X denotes GDP composition by agriculture which is X percent.

- When the GDP per capita income is high (40 instances), the GDP composition by agriculture is between 0 to 10 percent (39 instances) with a confidence of 0.98.
- When the GDP composition by services is between 70 to 80 percent (32 instances), the GDP composition by agriculture is between 0 to 10 percent (29 instances) with a confidence of 0.91.

If the same data is allowed to undergo EM clustering using the Step 6, the visualizations (shown in Figure 4) that are obtained also prove this fact.

Then we queried the World FactBook database for literacy rate, labor force in agriculture, labor force in industry, and labor force in services of every country. Then using step 4, which is segmentation, we segmented each of the attributes of labor force in agriculture, labor force in industry, labor force in services into three classes namely low, medium, and high respectively. We segmented the literacy rate attribute into three classes namely low, medium, and high depending on whether the literacy rate is between 0 and 50, 50 to 85, and 85 to 100 respectively. Here we are comparing the effects of labor force on each sector on the literacy rate of the country. Figure 5 shows the effect of labor force from agriculture on literacy rate.

We have also tested our tool by retrieving information about movies from 1991 to 2001 by DBpedia and Linked Movie Data Base from various countries and integrated that with data retrieved from the World FactBook like median age of the population and total population and found out the following patterns.

- If the population is greater than 58,147,733 and median age is greater than 38, the movie production is high with a confidence of 1.

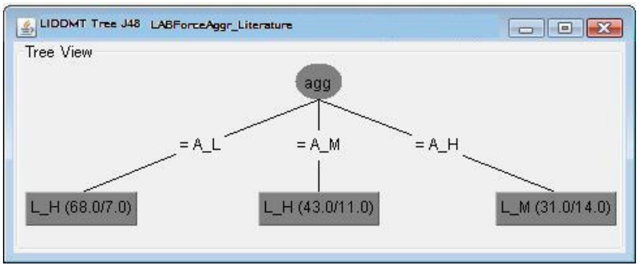


Figure 5: This figure shows that when labor force from agriculture is low (A_L), then literacy rate is high (L_H) with a 7 percent error rate out of 68 instances. Also when labor force from agriculture is medium (A_M), then the literacy rate is high (L_H) with 11 percent error rate out of 43 instances. Thus this can signify an inverse relationship between literacy rate and labor force in agriculture.

- If population is between 58,147,733 and 190,010,647, and median age is less than 38, the movie production is low with a confidence of 1.

Thus the above results prove that our LiDDMT is helping us to find out hidden relationships between the attributes in linked data thereby helping effectively in Knowledge Discovery.

6. DISCUSSIONS AND FUTURE WORK

From our experiments and case study we can say that the model that we proposed, LiDDM, has its strength in that it can retrieve data from multiple data sources and integrate them instead of just retrieving the data from a single data source. It can treat data from various sources in the same manner. The preprocessing and transformation steps make our model unique to deal with linked data. This allows us the flexibility of choosing data at will and then concentrates on mining. Also our tool, LiDDMT, helps us to mine and visualize data from more than one ARFF file at the same time, thus giving us the option for comparison.

By introducing graph-based techniques, triples could be found out automatically in future. Also, currently all the available predicates are obtained for only DBpedia and Linked Movie Data Base. For others you have to specify the predicates yourselves without prefixes if you use the automatic query builder. This functionality can be extended to other data sources easily. Thus, more and more data sets can be implemented here drawing predicates from all of them. But with our tool, even though you cannot get all the available predicates for datasets other than DBpedia and Linked Movie Data Base, you can use the automatic query builder to generate SPARQL queries automatically, if you know the URI of the predicate that you are using. Thus, more functionality can be imparted into the automatic query builder.

Also in future, some artificial intelligence measures can be introduced into LiDDM for suggesting the best machine learning algorithms that can give the best possible results depending on the data obtained from the linked data cloud. All in all, the existing functionality of the LiDDMT has been tested with many examples and our tool is proved to be very effective and usable.

7. CONCLUSIONS

Linked data with all its diversity and complexity acts as a huge database of information in RDF format, which is machine readable. There is a need to mine that data to find different hidden patterns and also make it conceivable for people to find out what it has in store for us.

Our model, LiDDM, successfully builds a data mining mechanism on top of linked data for effective understanding and analysis of linked data. The features in our model are built upon the classical KDD process and are modified to serve the needs of linked data. The step of getting the required data from the remote database itself makes our model dynamic. Flexibility is an added feature of our model as the steps of data retrieval and mining are separate. This allows users to retrieve all the possible results first and then to decide on the mining techniques. Also, the smooth cyclic movement in Step 3, Step 4, and Step 5, i.e. filter, segmentation, and writer respectively, makes our model more adaptable and more inclined towards removal of unwanted data and finding richer patterns. Visualizations at the end solve our problem by pictorially representing the interesting relationships hidden in the data there by making the data more understandable.

Regarding our tool, LiDDMT which we built on top of our model, the functioning is effective and the results are efficient as shown in case studies. Using Weka in our tool for the process of data mining makes it more efficient considering the vast popularity of Weka. The tool has much functionality implemented at each step of our model in an effort to make it more dynamic and usable. Also, having a chance to view more than one visualization at a time when implementing more than one data mining method makes our tool a very suitable one to compare data. But still the tool could be made more efficient as we discussed in the previous section.

8. REFERENCES

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference On Very Large Data Bases*, pages 487–499, San Francisco, Ca., USA, Sept. 1994. Morgan Kaufmann Publishers, Inc.

[2] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. DBpedia: A nucleus for a web of open data. In *Proceedings of the 6th International Semantic Web Conference*, volume 4825 of *Lecture Notes in Computer Science*, pages 722–735, Busan, Korea, Nov. 2007.

[3] C. Bizer, T. Heath, and T. Berners-Lee. Linked data - the story so far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22, 2009.

[4] J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson. Jena: implementing the semantic web recommendations. In *Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters*, pages 74–83, New York, NY, USA, 2004. ACM.

[5] Central Intelligence Agency. The world factbook. https://www.cia.gov/library/publications/the-world-factbook/,2011.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[7] J. Dokulil and J. Katreniaková. RDF query generator. In *Proceedings of the 12th International Conference on Information Visualisation*, pages 191–193. IEEE Computer Society, 2008.

[8] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39(11):27–34, Nov. 1996.

[9] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA data mining software: an update. *SIGKDD Explorations*, 11(1):10–18, 2009.

[10] O. Hassanzadeh and M. Consens. Linked movie data base. In *Proceedings of the Linked Data on the Web Workshop*, 2009.

[11] T. Heath. How will we interact with the web of data? *IEEE Internet Computing*, 12(5):88–91, 2008.

[12] C. Kiefer, A. Bernstein, and A. Locher. Adding data mining support to SPARQL via statistical relational learning methods. In *Proceedings of the 5th European Semantic Web Conference*, volume 5021 of *Lecture Notes in Computer Science*, pages 478–492. Springer, 2008.

[13] Machine Learning Group at University of Waikato. Attribute-relation file format. http://www.cs.waikato.ac.nz/ ml/weka/arff.html, 2008.

[14] V. Nebot and R. Berlanga. Mining association rules from semantic web data. In *Proceedings of the 23rd International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems*, volume 6097 of *Lecture Notes in Computer Science*, pages 504–513. Springer Berlin / Heidelberg, 2010.

[15] J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo, CA, 1993.

[16] the United States Government. Data.gov. http://www.data.gov/, 2011.

[17] C. Wastyn. Drupal sparql query builder. http://drupal.org/node/306849, 2008.

[18] E. PrudŠhommeaux and A. Seaborne. SPARQL Query Language for RDF. In *W3C WD*,4th October, 2006. http://www.w3.org/TR/2006/WD-rdf-sparql-query-20061004